Arh

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/704,186	10/31/2000	Bhalchandra S. Pandit	MS1-611US	3152

22801 7590 04/22/2004

LEE & HAYES PLLC
421 W RIVERSIDE AVENUE SUITE 500
SPOKANE, WA 99201

EXAMINER	
WU, ALLEN S	
ART UNIT	PAPER NUMBER
2135	6

DATE MAILED: 04/22/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/704,186	PANDIT ET AL.	
	Examiner	Art Unit	
	Allen S. Wu	2135	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *31 October 2000*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-40* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-40* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *31 October 2000* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)

2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date 4.

4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .

5) ☐ Notice of Informal Patent Application (PTO-152)

6) ☐ Other: _____.

DETAILED ACTION

Claim Rejections - 35 USC § 112

1. The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

2. Claim 37 is rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

Claim 37 claims a resource manager that is not part of an operating system. The specification fails to disclose a resource manager that is not part of an operating system for such a system of the claimed invention.

Claim Rejections - 35 USC § 103

3. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

4. Claims 1-7 are rejected under 35 U.S.C. 103(a) as being unpatentable over Baclawski, US Patent 6,505,191, in view of Chen et al (hereinafter Chen), US Patent 5,852,821, and further in view of Applicant's Admitted Prior Art (hereinafter AAPA).

As per claim 1, Baclawski discloses a means of matching an input value with a target value (see for example; search engine, abstract) including; separating the input into a plurality of portions (see for example; col 8 ln 4-10); indexing into each of a plurality of sub-hashes (index nodes) using of the plurality of portions (see for example; col 9 ln 26-43) and combining the plurality of values to generate a result of the match (see for example, col 9 ln 51-56). As for generating a hash key based on the input value and separating the hash key into a plurality of portions, Baclawski discloses an alternative means wherein the input value is separated into fragments and each of the fragments are hashed (see for example; col 8 ln 4-9 and ln 61-67). Baclawski uses this method as a means of indexing into a plurality of sub-hashes (data in local database). It would have been obvious to one of ordinary skill in the art at the time of the applicant's invention to have a design choice in generating a hash key on the input value and separating the hash key into a plurality of portions because the applicant has not explicitly stated any other reason or purpose for such means other than for generating an index to lookup values and the means disclosed by Baclawski is just as efficient.

Baclawski does not explicitly teach combining a plurality of values to generate a hash result and comparing the input value to corresponding target value to determine whether the values match. Chen discloses a second means of matching input value with a target value wherein a plurality of values are combined to generate a result (see for example; col 5 ln 21-30), wherein each bit

in the result corresponds to one of a plurality of target values (see for example; col 6 ln 7-30) and performs operations for each bit in the result that is set (see for example; col 6 ln 8-13). Both Chen and Baclawski disclose a means of searching a database quickly and efficiently ending in different results. Chen further discloses use of such bit comparison to decrease memory access (see for example; col 6 ln 24-31). It would have been obvious to one of ordinary skill in the art at the time of the applicant's invention to combine the teachings of Chen within the system of Baclawski because it would have provided an increase in efficiency through less memory accesses to respond to a query.

Furthermore, the Baclawski-Chen does not explicitly teach comparing the input value to the corresponding target value to determine whether the values match. AAPA discloses the use of a result to determine which target values is likely matches (see for example; page 2 ln 5-17). Chen discloses the use of a result as a memory look up, one of ordinary skill in the art at t the time of the applicant's invention would have been able to modify the lookup with a further comparison to check for an actual match. Performing such comparison before reporting search results is well known in the art to reduce error in searches. It would have been obvious to one of ordinary skill in the art to combine AAPA within the Baclawski-Chen combination because it would have reduced the error of search queries by first comparing the target values with the input value to determine a match.

As for a plurality of sub-hashes and a hash result, Baclawski discloses the use of hashes to index through data as described above. Chen discloses the use of sub-entries for to produce a result. One of ordinary skill in the art at the time of the applicant's invention would have recognized the sub-hashes (sub-entry of Chen) and hash result (result of Chen) through the combination of Baclawski and Chen.

As per claim 2, Baclawski-Chen-AAPA discloses the claim limitations as described above (see claim 1). As per the number of target values in the plurality of target values being equal to the number of bits in the hash bits in the hash result, one of ordinary skill in the art would have recognized the amount of target values must be mapped to the number of bits in the result in the Baclawski-Chen-AAPA combination. If the number of target values is not equal to the number of bits in a result, then a mapping to the target values from the result will not exist. Since each bit corresponds to a target value, one of ordinary skill in the art would have recognized that the number of target values must be equal to the number of bits in the result for such a mapping to exist.

As per claim 3, Baclawski-Chen-AAPA discloses the claim limitations as described above (see claim 1). As per the number of target values in the plurality of target values being equal to the number of bits in the hash bits in the hash result, one of ordinary skill in the art would have recognized the amount of target

values must be mapped to the number of bits in the result in the Baclawski-Chen-AAPA combination. If the maximum number of target values is not equal to the number of bits in a result, then a mapping to the target values from the result will not exist. Since each bit corresponds to a target value, one of ordinary skill in the art would have recognized that the maximum number of target values must be equal to the number of bits in the result for such a mapping to exist.

As per claim 4, Baclawski-Chen-AAPA discloses the claim limitations as described above (see claim 1). As for a maximum number of target values in the plurality of target values being equal to the number of bits in each of a plurality of locations of the plurality of sub-hashes that can be indexed, Chen discloses accessing a number of bit vectors and combining relevant bit vectors to create a result (see for example; col 5 ln 45-col 6 ln 7) and wherein the result is a bit mapping to target values (see for example; col 6 ln 8-30). Since each bit corresponds to a target value, one of ordinary skill in the art would have recognized that the maximum number of target values must be equal to the number of bits in the result for such a mapping to exist. Furthermore, in order to achieve such result, the bit vectors combined through logical operations must be the same number of bits as the result for such operations to be implementable. For example, a bit vector with 5 bits cannot be logically ANDed together with a bit vector of 3 bits. Therefore, one of ordinary skill in the art would have recognized that a maximum number of target values in the plurality of target values being

equal to the number of bits in each of a plurality of locations of the plurality of sub-hashes that can be indexed must exist for such combining of sub-hashes to create a result having the correspond number of bits as the maximum number of target values.

As per claim 5, Baclawski-Chen-AAPA discloses the claim limitations as described above (see claim 1). Baclawski discloses separating a value into two portions (see for example; fragments, col 8 ln 4-9; there can be any amount of plurality of fragments depending the input value, including two).

As per claim 6, Baclawski-Chen-AAPA discloses the claim limitations as described above (see claim 1). Baclawski further discloses separating an input value into a plurality of contiguous and equal portions (see for example; col 8 ln 49-60).

As per claim 7, Baclawski-Chen-AAPA discloses the claim limitations as described above (see claim 1). Chen further discloses combining comprising of bit wise logical ANDing of the plurality of values (see for example; col 5 ln 45=col 6 ln 7).

5. Claims 8-18 are rejected under 35 U.S.C. 103(a) as being unpatentable over Chen et al (hereinafter Chen), US Patent 5,852,821 in view of Srinivasan et al (hereinafter Srinivasan), US Patent 5,852,822.

As per claim 8, Chen discloses a plurality of sub-hashes (bit vector; see for example; col 4 ln 31-43); a plurality of indexes, each index being generated from a value (see for example; col 4 ln 31-43); and a combiner coupled to receive values from the plurality of sub-hashes (bit vector) on the plurality of indexes (see for example; col 5 ln 45-col 6 ln 7), and to generate a result by combining the received values (see for example; resultant bit vector, col 6 ln 8-13).

As for generation of a hash result using sub-hashes, hashing is well known in the art to be defined as a mapping of numerical values to convert a key or identifier into a corresponding location. Chen discloses the use of the bit vector to be a mapping to target values in memory (see for example; col 6 ln 14-29). One of ordinary skill in the art would have recognized the resultant bit vector as a hash result and the bit vectors used for generating such a resultant bit vector as the sub-hashes.

Chen does not explicitly teach the index being generated from a hash key, Srinivasan discloses indexes generated from a hash key and the index used to index into a plurality of data (see for example; col 9 ln 18-32). Hash keys are well known in the art to be used to quickly index into a plurality of data in a quick and efficient manner. . Srinivasan discloses the use of keys to map to an entry in an index. From the discussion of the definition of hash above, one of ordinary

skill in the art would have recognized the keys and sub-keys of Srinivasan to be hash key and sub-hash keys. It would have been obvious to one of ordinary skill in the art of at the time of the applicant's invention to combine the teachings of Srinivasan within the teachings of Chen because it would have speed up the processing and lookup of sub-hashes (bit vectors) in the index and thus increase processing efficiency and speed.

As per claim 9, Chen-Srinivasan discloses the claim limitations as described above (see claim 8) and further discloses a combinatorial logic component to perform a bit wise logical ANDing of the values received from the plurality of sub-hashes (see for example, col 5 ln 45-col 6 ln 7).

As per claim 10, Chen-Srinivasan discloses the claim limitations as described above (see claim 8) and further discloses the hashing architecture being implemented in software.

As per claim 11, Chen-Srinivasan discloses the claim limitations as described above (see claim 8). As for the hashing architecture being implemented in firmware, Chen further discloses the hashing architecture being implemented in software. It would have been obvious to one of ordinary skill in the art at the time of the applicant's invention to have a design choice of implementing the hashing architecture in firmware because the applicant' has not

explicitly stated any particular reason or purpose for such means other than to implement the hashing architecture, and the means of Chen is just as efficient.

As per claim 12, Chen-Srinivasan discloses the claim limitations as described above (see claim 8) and further discloses the hashing architecture being implemented in hardware (see for example; col 4 ln 17-20).

As per claim 13, Chen-Srinivasan discloses the claim limitations as described above (see claim 8). Srinivasan further discloses a generation of such indexes by separating the hash into a plurality of portions (see for example, col 9 ln 18-26). Srinivasan is silent on separating the hash key into a plurality of equal portions. It would have been obvious to one of ordinary skill in the art at the time of the applicant's invention to have a design choice in separating the hash key into a plurality of equal portion because the applicant has not explicitly stated any particular reason or purpose for such means other than for generating indexes, and the means of the Chen-Srinivasan combination is just as efficient.

As per claims 14 and 18, Chen discloses identifying a plurality of values from a plurality of sub-hashes (see for example, col 5 ln 30-45) and generating a hash result based on the plurality of values (see for example; col 6 ln 8-14).

As for generation of a hash result using sub-hashes, hashing is well known in the art to be defined as a mapping of numerical values to convert a key

or identifier into a corresponding location. Chen discloses the use of the bit vector to be a mapping to target values in memory (see for example; col 6 ln 14-29). One of ordinary skill in the art would have recognized the resultant bit vector as a hash result and the bit vectors used for generating such a resultant bit vector as the sub-hashes.

Chen does not explicitly teach generating a plurality of sub-hash keys based on a hash key, Srinivasan discloses sub-keys generated from a key and indexing into a plurality of sub-entries using the keys (see for example; col 9 ln 18-32). Hash keys are well known in the art to be used to quickly index into a plurality of data in a quick and efficient manner. Srinivasan discloses the use of keys to map to an entry in an index. From the discussion of the definition of hash above, one of ordinary skill in the art would have recognized the keys and sub-keys of Srinivasan to be hash key and sub-hash keys. It would have been obvious to one of ordinary skill in the art of at the time of the applicant's invention to combine the teachings of Srinivasan within the teachings of Chen because it would have speed up the processing and lookup of sub-hashes (bit vectors) in the index and thus increase processing efficiency and speed.

As per claim 15, Chen-Srinivasan discloses the claim limitations as described above (see claim 14), and further discloses generating the hash key prior to generating the plurality of sub-hash keys (see for example; col 9 ln 18-32; a key must be generated before it can be divided).

As per claim 16, Srinivasan further discloses a generation of such indexes by separating the hash into a plurality of equal portions (see for example, col 9 ln 18-26). Srinivasan is silent on separating the hash key into a plurality of equal portions. It would have been obvious to one of ordinary skill in the art at the time of the applicant's invention to have a design choice in separating the hash key into a plurality of equal portion because the applicant has not explicitly stated any particular reason or purpose for such means other than for generating indexes, and the means of the Chen-Srinivasan combination is just as efficient.

As per claim 17, Chen-Srinivasan disclose the claim limitations as described above (see claim 14), and further discloses a combinatorial logic component to perform a bit wise logical ANDing of the values received from the plurality of sub-hashes (see for example, col 5 ln 45-col 6 ln 7).

6. Claims 19-37 are rejected under 35 U.S.C. 103(a) as being unpatentable over Applicant's Admitted Prior Art (hereinafter, AAPA), in view of Chen et al (hereinafter Chen), US Patent 5,852,821 and further in view of Srinivasan et al (hereinafter Srinivasan), US Patent 5,852,822.

As per claim 19, AAPA discloses a means of determining whether a security identifier of an access control element matches any of a plurality of security identifiers of a security token (see for example, page 7 ln 1-7) comprising

generating a hash key based on the access control element security identifier (see for example, page 2 ln 5-7) and indexing into hash table to obtain a result that indicates which security identifier corresponding to the user are likely matches (see for example, page 2 ln 7-10).

AAPA does not explicitly teach a first and second sub-hash. Chen discloses a means of comparing values including the use of a first and second sub-hash (bit-vectors) wherein the sub-hash (bit vectors) are combined to form a result (see for example; col 5 ln 45-col 6 ln 7), wherein each bit in the result value corresponds to one of the plurality of target values (see for example; col 6 ln 8-13). One of ordinary skill in the art at the time of the applicant's invention would have recognized that the sub-hashing of Chen would could have been used to obtain a hash result and that each of the bits in the hash result of Chen is mapped to a corresponding one of the plurality of security token security identifiers of AAPA. It would have been obvious to one of ordinary skill in the art to combine the teachings of Chen within the system of AAPA because it would have increased processing time and efficiency to the indexing means of AAPA (see for example; Chen, col 2 ln 35-49).

As for generation of a hash result using sub-hashes, hashing is well known in the art to be defined as a mapping of numerical values to convert a key or identifier into a corresponding location. Chen discloses the use of the bit vector to be a mapping to target values in memory (see for example; col 6 ln 14-29). One of ordinary skill in the art would have recognized the resultant bit vector

as a hash result and the bit vectors used for generating such a resultant bit vector as the sub-hashes.

AAPA-Chen does not explicitly teach the indexing into sub-hashes using a first and second sub-hash key. Srinivasan discloses indexes generated from a hash key and the hash key separated into a first and second portion (see for example; col 9 ln 18-36) used to index into a first and second portion of data (see for example; col 9 ln 18-32 and col 10 ln 7-19). Hash keys are well known in the art to be used to quickly index into a plurality of data in a quick and efficient manner. Srinivasan discloses the use of keys to map to an entry in an index. From the discussion of the definition of hash above, one of ordinary skill in the art would have recognized the keys and sub-keys of Srinivasan to be hash key and sub-hash keys. It would have been obvious to one of ordinary skill in the art of at the time of the applicant's invention to combine the teachings of Srinivasan within the AAPA-Chen combination because it would have speed up the processing and lookup of sub-hashes (e.g. bit vectors) in the index and thus increase processing efficiency and speed.

As per claim 20, AAPA-Chen-Srinivasan discloses the claim limitations as described above (see claim 19). AAPA further discloses that keys can be generated in any of a wide variety of conventional manners (see for example, AAPA page 8 ln 2-3). It would have been obvious to one of ordinary skill in the art at the time of the applicant's invention to have a design choice in generating a

hash key by selecting a portion of the access control element security identifier because the applicant has not explicitly stated any particular reason or purpose for such selection and generation means other than for generating a hash key and any convention manners of AAPA-Chen-Srinivasan is just as efficient.

As per claim 21, AAPA-Chen-Srinivasan discloses the claim limitations as described above (see claim 19). Srinivasan further discloses a generation of such indexes by separating the input into a plurality of contiguous portions (see for example, col 9 ln 18-26). Srinivasan is silent on separating the hash key into a plurality of equal portions. It would have been obvious to one of ordinary skill in the art at the time of the applicant's invention to have a design choice in separating the hash key into a plurality of equal portion because the applicant has not explicitly stated any particular reason or purpose for such means other than for generating indexes, and the means of the AAPA-Chen-Srinivasan combination is just as efficient.

As per claim 22, AAPA-Chen-Srinivasan discloses the claim limitations as described above (see claim 19). Chen further discloses a bit wise logical ANDing of the values received from the plurality of sub-hashes (see for example, col 5 ln 45-col 6 ln 7).

As per claims 23 and 30, AAPA discloses a means of determining whether a security identifier of an access control element matches any of a plurality of security identifiers of a security token (see for example, page 7 ln 1-7) comprising generating a hash key based on the access control element security identifier (see for example, page 2 ln 5-7) and indexing into hash table to obtain a result that indicates which security identifier corresponding to the user are likely matches (see for example, page 2 ln 7-10).

AAPA does not explicitly teach a plurality of sub-hashes. Chen discloses a means of comparing values including combining the plurality of values of a plurality of sub-hashes (bit vector) to form a result (see for example; col 5 ln 45-col 6 ln 7), wherein each bit in the result value corresponds to one of the plurality of target values (see for example; col 6 ln 8-13). One of ordinary skill in the art at the time of the applicant's invention would have recognized that the sub-hashing of Chen would could have been used to obtain a hash result and that each of the bits in the hash result of Chen is mapped to a corresponding one of the plurality of security token security identifiers of AAPA. It would have been obvious to one of ordinary skill in the art to combine the teachings of Chen within the system of AAPA because it would have increased processing time and efficiency to the indexing means of AAPA (see for example; Chen, col 2 ln 35-49).

As for generation of a hash result using sub-hashes, hashing is well known in the art to be defined as a mapping of numerical values to convert a key or identifier into a corresponding location. Chen discloses the use of the bit

vector to be a mapping to target values in memory (see for example; col 6 ln 14-29). One of ordinary skill in the art would have recognized the resultant bit vector as a hash result and the bit vectors used for generating such a resultant bit vector as the sub-hashes.

AAPA-Chen does not explicitly teach the indexing into sub-hashes using a respective one of the plurality of sub-hash indexes. Srinivasan discloses indexes generated from a hash key and the hash key separated into a plurality of indexes (see for example; col 9 ln 18-36) and is used to index a respective portion of data (see for example; col 9 ln 18-32 and col 10 ln 7-19). Hash keys are well known in the art to be used to quickly index into a plurality of data in a quick and efficient manner. Srinivasan discloses the use of keys to map to an entry in an index. From the discussion of the definition of hash above, one of ordinary skill in the art would have recognized the keys and sub-keys of Srinivasan to be hash key and sub-hash keys. It would have been obvious to one of ordinary skill in the art of at the time of the applicant's invention to combine the teachings of Srinivasan within the AAPA-Chen combination because it would have speed up the processing and lookup of sub-hashes (e.g. bit vectors) in the index and thus increase processing efficiency and speed.

As per claim 24, AAPA-Chen-Srinivasan discloses the claim limitations as described above (see claim 23). Chen further discloses wherein the particular value comprises a value of one (see for example, col 6 ln 10).

As per claim 25, AAPA-Chen-Srinivasan discloses the claim limitations as described above (see claim 23). Chen further discloses wherein each portion is a bit of the result value (see for example; col 6 ln 8-13).

As per claim 26, AAPA-Chen-Srinivasan discloses the claim limitations as described above (see claim 23). Srinivasan further discloses a generation of such indexes by separating the input into a plurality of contiguous portions (see for example, col 9 ln 18-26). Srinivasan is silent on separating the hash key into a plurality of equal portions. It would have been obvious to one of ordinary skill in the art at the time of the applicant's invention to have a design choice in separating the hash key into a plurality of equal portion because the applicant has not explicitly stated any particular reason or purpose for such means other than for generating indexes, and the means of the AAPA-Chen-Srinivasan combination is just as efficient.

As for using a using a portion of the security identifier, AAPA further discloses that keys can be generated in any of a wide variety of conventional manner (see for example, AAPA page 8 ln 2-3). It would have been obvious to one of ordinary skill in the art at the time of the applicant's invention to have a design choice in generating a hash key by selecting a portion of the access control element security identifier because the applicant has not explicitly stated any particular reason or purpose for such selection and generation means other

than for the means of indexing into sub-hashes and any convention manners of AAPA-Chen-Srinivasan is just as efficient.

As per claim 27, AAPA-Chen-Srinivasan discloses the claim limitations as described above (see claim 23). Chen further discloses a bit wise logical ANDing of the values received from the plurality of sub-hashes (see for example, col 5 ln 45-col 6 ln 7).

As per claim 28, AAPA-Chen-Srinivasan discloses the claim limitations as described above (see claim 19). AAPA further discloses comparing a security token security identifier with an access control security identifier (see for example, fig 1 and page 1 ln 13-21) and that such searching comprises a linear search (see for example; page 1 ln 22-23). One of ordinary skill in the art at the time of the applicant's invention would have recognized that such a comparing would entail inputting security identifier being a security token security identifier and the plurality of target security identifiers comprising an access control security identifier in the AAPA-Chen-Srinivasan combination. Furthermore, the since the search is linear, input into the search and target identifiers can be reversed and obtain the same result. Therefore, it would have been obvious to one of ordinary skill in the art at the time of the applicant's invention to have a design choice in inputting a access control security identifier and a plurality of target security identifiers being a security token security identifier because the

applicant has not explicitly stated any particular reason or purpose for such inputting and targeting other than for implementing access control on a computer system, and the means of inputting and target values of AAPA-Chen-Srinivasan is just as efficient.

As per claim 29, AAPA-Chen-Srinivasan discloses the claim limitations as described above (see claim 19). AAPA further discloses comparing a security token security identifier with an access control security identifier (see for example, fig 1 and page 1 ln 13-21) and that such searching comprises a linear search (see for example; page 1 ln 22-23). One of ordinary skill in the art at the time of the applicant's invention would have recognized that such a comparing would entail inputting security identifier being a security token security identifier and the plurality of target security identifiers comprising an access control security identifier in the AAPA-Chen-Srinivasan combination.

As per claim 31, AAPA discloses a plurality of security token identifiers corresponding to a user (see for example, page 1 ln 13-21 and page 7 ln 6-7), a plurality of access control security identifiers corresponding to an object (see for example, page 1 ln 13-21 and page 7 ln 6-7) an access controller to determine whether any of the plurality of security token security identifiers match any of the plurality of access control security identifiers (see for example, page 1 ln 17-21 and page 7 ln 6-7; and access controller must be present for such a

determination. AAPA further discloses generating a hash key based on the access control element security identifier (see for example, page 2 ln 5-7) and indexing into hash table to obtain a result that indicates which security identifier corresponding to the user are likely matches (see for example, page 2 ln 7-10).

AAPA does not explicitly teach a plurality of sub-hashes. Chen discloses a means of comparing values including the use of a first and second sub-hash (bit-vectors) wherein the sub-hash (bit vectors) are combined to form a result (see for example; col 5 ln 45-col 6 ln 7), wherein each bit in the result value corresponds to one of the plurality of target values (see for example; col 6 ln 8-13). One of ordinary skill in the art at the time of the applicant's invention would have recognized that the sub-hashing of Chen would could have been used to obtain a hash result and that each of the bits in the hash result of Chen is mapped to a corresponding one of the plurality of security identifiers of AAPA. It would have been obvious to one of ordinary skill in the art to combine the teachings of Chen within the system of AAPA because it would have increased processing time and efficiency to the indexing means of AAPA (see for example; Chen, col 2 ln 35-49).

As for generation of a hash result using sub-hashes, hashing is well known in the art to be defined as a mapping of numerical values to convert a key or identifier into a corresponding location. Chen discloses the use of the bit vector to be a mapping to target values in memory (see for example; col 6 ln 14-29). One of ordinary skill in the art would have recognized the resultant bit vector

as a hash result and the bit vectors used for generating such a resultant bit vector as the sub-hashes.

AAPA-Chen does not explicitly teach the indexing into sub-hashes using a respective one of a plurality of sub-hash keys. Srinivasan discloses indexes generated from a hash key and the hash key separated into a plurality of portions (see for example; col 9 ln 18-36) used to index into respective parts of data (see for example; col 9 ln 18-32 and col 10 ln 7-19). Hash keys are well known in the art to be used to quickly index into a plurality of data in a quick and efficient manner. Srinivasan discloses the use of keys to map to an entry in an index. From the discussion of the definition of hash above, one of ordinary skill in the art would have recognized the keys and sub-keys of Srinivasan to be hash key and sub-hash keys. It would have been obvious to one of ordinary skill in the art of at the time of the applicant's invention to combine the teachings of Srinivasan within the AAPA-Chen combination because it would have speed up the processing and lookup of sub-hashes (e.g. bit vectors) in the index and thus increase processing efficiency and speed.

As per claim 32, AAPA-Chen-Srinivasan discloses the claim limitations as described above (see claim 31). Chen further discloses the number of sub - hashes depend on the number of bit patterns, therefore the sub-hashes are not limited in the amount. It would have been obvious to one of ordinary skill in the art at the time of the applicant's invention to have a design choice in the number

of sub-hashes being two since the applicant has not explicitly stated any particular reason or purpose for such a number of sub-hashing other than for mapping to a hash result and the means of AAPA-Chen-Srinivasan is just as efficient.

As per claim 33, AAPA-Chen-Srinivasan discloses the claim limitations as described above (see claim 31). Chen further discloses each bit in the result value that is set has a value of one (see for example, col 6 ln 10).

As per claim 34, AAPA-Chen-Srinivasan discloses the claim limitations as described above (see claim 31). Chen further discloses a bit wise logical ANDing of the values received from the plurality of sub-hashes (see for example, col 5 ln 45-col 6 ln 7).

As per claim 35, AAPA-Chen-Srinivasan discloses the claim limitations as described above (see claim 31). As per the number of target values in the plurality of target values being equal to the number of bits in the hash bits in the hash result, one of ordinary skill in the art would have recognized the amount of target values must be mapped to the number of bits in the result in the AAPA-Chen-Srinivasan combination. If the maximum number of target values (security token security identifiers) is not equal to the number of bits in a result, then a mapping to the target values from the result will not exist. Since each bit

corresponds to a target value, one of ordinary skill in the art would have recognized that the maximum number of target values must be equal to the number of bits in the result for such a mapping to exist.

As per claim 36, AAPA-Chen-Srinivasan discloses the claim limitations as described above (see claim 31). AAPA further discloses an access control system incorporated through an operating system (see for example; page 1 ln 6-12).

As per claim 37, AAPA-Chen-Srinivasan discloses the claim limitations as described above (see claim 31). AAPA further discloses an access control system incorporated through an operating system (see for example; page 1 ln 6-12). It would have been obvious to one of ordinary skill in the art at the time of the applicant's invention to have a design choice in using a resource manager that is not part of an operating system because the applicant has not explicitly stated any particular reason or purpose for such a manger other than for system operating means and the means of AAPA-Chen-Srinivasan is just as efficient.

7. Claims 38-40 are rejected under 35 U.S.C. 103(a) as being unpatentable over of Chen et al (hereinafter Chen), US Patent 5,852,821.

As per claims 38 and 40, Chen discloses for each sub-hash in a plurality of sub-hashes that can be used together to generate a hash result (see for example col 5 ln 45-col 6 ln 13),

(a) identifying a bit in a location of a sub-hash (see for example; col 4 ln 30-36),

(b) identifying in a source value, a plurality of bits corresponding to the sub-hash (see for example, col 4 ln 55-57),

(c) comparing the identifier of the location to a plurality of bits (see for example; col 4 ln 57-63)

(d) setting the bit if the identifier of the location matches the plurality of bits (see for example, col 4 ln 30-43) or clearing the bit (see for example col 4 ln 51-52; all bits are cleared beforehand), and

(e) repeating acts (a), (b), (c), and (d) for each of a plurality of bits in the location of the sub-hash (see for example, col 4 ln 56-64; each bit is set according to a location).

As for hash and sub-hashes, hashing is well known in the art to be defined as a mapping of numerical values to convert a key or identifier into a corresponding location. Chen discloses the use of the bit vector to be a mapping to target values in memory (see for example; col 6 ln 14-29). One of ordinary skill in the art would have recognized the resultant bit vector as a hash result and the bit vectors used for generating such a resultant bit vector as the sub-hashes.

As per claim 39, Chen further discloses the plurality of bits correspond to a part of a portion of the source value that will be used to generate a hash value (resultant bit; see for example, col 6 ln 8-13).

Conclusion

8. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

US Patent 6,424,650 discloses using a hash value to index into a table.

US Patent 5,742,807 discloses use of multiple hash table as a lookup.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Allen S. Wu whose telephone number is 703-305-0708. The examiner can normally be reached on Monday-Friday 9am-5pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Kim Vu can be reached on 703-305-4393. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Allen Wu
Patent Examiner
Art Unit 2135

ASW

KIM VU
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100